- 9 -

REMARKS

The Examiner has maintained the rejection of the claims. As set forth below, such rejection is still deficient. However, despite such deficiencies and in the spirit of expediting the prosecution of the present application, applicant has incorporated the subject matter of at least one dependent claim into each of the independent claims. Since the subject matter of such dependent claim(s) was already considered by the Examiner, it is asserted that such claim amendments would **not** require new search and/or consideration.

The Examiner has remarked that the status identifier for Claim 2 should be "previously presented." Applicant has changed such status identifier accordingly.

The Examiner has rejected Claims 13-14, 28-29 and 43-44 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, the Examiner has stated that with respect to Claims 13, 18 and 43, applicant has duplicated claim language in lines 9-10 of each of such claims. This rejection is deemed moot in view of the claim amendments made hereinabove.

The Examiner has rejected Claims 1-2, 4-6, 9-14, 16-17, 19-21, 24-29, 31-32, 34-36, 39-44 and 46-48 under 35 U.S.C. 103(a) as being unpatentable over Hammack (U.S. Patent No. 6,449,624) in view of Berg (U.S. Patent No. 6,745,208). Applicant respectfully disagrees with such rejection, especially in view of the amendments made hereinabove with respect to independent Claims 1, 16 and 31.

With respect to independent Claims 1, 16 and 31, the Examiner has relied on Col. 3, lines 41-46; Col. 12, lines 25-36; Col. 14, lines 34-47; and Col. 21, lines 33-49 in Hammack to make a prior art showing of applicant's claimed "comparing code operable to compare an XML data representation of said program configuration data with data defining valid program configuration data; wherein, if said XML data representation does

- 10 -

match said data defining valid program configuration data, then triggering code is operable to trigger a valid program configuration response" (see this or similar, but not necessarily identical language in the foregoing claims). In the latest Office Action dated 12/23/2005, the Examiner has responded to applicant's arguments by stating that the "prior configuration version [in Hammack] is interpreted as the data defining valid program configuration data."

Applicant respectfully disagrees. In particular, Hammack teaches "recording data indicative of modifications of the configuration of a process" where such "data is stored...as versions" and that "a user [can] compare the differences between any two versions and return the process to a desired prior configuration" (see Col. 3, lines 37-47). Thus, in Hammack, a user can compare the differences between any two versions that have been recorded, where such versions relate to instances where modifications have been made to the configuration. Since recorded modified versions, as in Hammack, does not even suggest whether any of such modified versions are valid, Hammack therefore cannot meet applicant's claimed "valid program configuration data" (emphasis added).

Applicant also respectfully asserts that applicant's claimed technique "wherein, if said XML data representation does match said data defining valid program configuration data, then triggering code is operable to trigger a valid program configuration response" further emphasizes the deficiencies in Hammack noted above. In particular, Hammack teaches "recording data indicative of modifications of the configuration" where such "data is stored...as versions" and "a user compare[s] the differences between any two versions." Since Hammack's disclosed versions each relate to instances where a modification is made to the configuration, none of Hammack's versions could be compared to create a match, whereas applicant specifically claims a situation where "said XML data representation does match said data defining valid program configuration data" (emphasis added).

With respect to each of the remaining independent claims, applicant notes that the Examiner has simply relied on the same excerpt as relied on in the last Office Action

- 11 -

dated 7/1/05 without responding to applicant's specific arguments in the Amendment
dated 10/12/05. In particular, the Examiner has again relied on Col. 21, lines 33-49 in
Hammack to make a prior art showing of applicant's claimed techniques "wherein
mapping code is operable to map between said program configuration data and said XML
data representation of said program configuration data" and "wherein editing code is
operable to edit said DOM data representation of said program configuration data to
provide modified program configuration data to be validated" (see the same or similar,
but not necessarily identical language in each of the independent claims).

Applicant again respectfully asserts that simply nowhere in the entire Hammack
reference, and especially not in the excerpt relied on by the Examiner, is there even a
suggestion of a map between program configuration data and an XML data representation
of the program configuration data or of editing code operable to edit the DOM data
representation of the program configuration data to provide modified program
configuration data to be validated, in the manner claimed by applicant. In particular,
applicant notes that Hammack only teaches that "version control data [may be stored] in a
file in accordance with...XML" and that "each configuration version record has a field
dedicated to having...text stored therein that represents the version control data
associated with the configuration version."

Thus, Hammack only teaches that each <u>versionable item</u> has a database record
with XML text representing version control data associated with the configuration
version. Hammack also teaches that such "versionable item should be understood to
include any item for which historical configuration information is maintained or stored <u>in</u>
<u>association with</u> the item itself" (emphasis added). Clearly, a versionable item associated
with configuration information does not meet a map between the actual "<u>program</u>
<u>configuration data</u> and [the] XML data representation of said program configuration
data," as applicant claims (emphasis added). In addition, the Examiner has stated that
Hammack teaches "editing code operable to edit said <u>XML</u> data representation of said
program configuration data" (emphasis added). Applicant respectfully points out that
what is claimed is "editing code operable to edit said <u>DOM</u> data representation of said

- 12 -

program configuration" (emphasis added). Hammack fails to even suggest DOM data, as applicant claims, let alone "edit[ing] said DOM data representation of said program configuration to provide modified program configuration data to be validated," as claimed (emphasis added).

Still with respect to each of the independent claims, the Examiner has relied on Col. 3, lines 44-57 and Col. 4, lines 17-24 in Berg when stating that Berg teaches a DOM data representation and editing a DOM data representation. Applicant assumes that the Examiner has relied on such excerpts to make a prior art showing of applicant's claimed technique "wherein said mapping code is operable to map between said program configuration data and said XML data representation via a DOM data representation of said program configuration data" since nowhere else in the Examiner's rejection has the Examiner specifically addressed such claim language.

Applicant respectfully asserts that Berg teaches "synchronizing an XML document object model (DOM) with the corresponding object model" (see Abstract) such that changes made to either the XML DOM or the object model are only reflected in each other without having to re-parse the entire XML file (see Col. 3, lines 25-34). Clearly, only synchronizing a DOM and an object model does not meet applicant's claimed "map between said program configuration data and said XML data representation" (emphasis added).

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. *In re Vaeck,* 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

- 13 -

Applicant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above. Nevertheless, despite such paramount deficiencies and in the spirit of expediting the prosecution of the present application, applicant has substantially incorporated the subject matter of Claims 9 and 10 et al. into independent Claims 1, 16 and 31.

With respect to the subject matter of Claim 9 et al., presently incorporated into independent Claims 1, 16 and 31, the Examiner has relied on Col. 14, lines 34-47 in Hammack to make a prior art showing of applicant's claimed technique "wherein said comparing code is executable by a program configuration managing computer and said valid program configuration response comprises sending validated program configuration data to a managed computer for use by said managed computer."

Applicant respectfully asserts that the excerpt relied on by the Examiner only discloses a "version audit report" by which "version identifying information may be made available to the process designers." Clearly ,sending version identifying information to a process designer, as in Hammack, does not meet applicant's claimed "comparing code [that is] is executable by a program configuration managing computer" (emphasis added). In addition, Hammack only generally discloses that version identifying information is made available, but does not specifically disclose that validated program configuration data is transferred in any way. Hammack also only teaches that such information is sent to a process designer, and not "to a managed computer for use by said managed computer," as applicant claims (emphasis added).

With respect to the subject matter of Claim 10 et al, presently incorporated into independent Claims 1, 16 and 31, the Examiner has relied on Col. 14, lines 34-47 and Col. 21, lines 33-49 in Hammack to make a prior art showing of applicant's claimed technique "wherein said validated program configuration data is sent from said program configuration managing computer to said managed computer as said XML data

- 14 -

representation." Applicant respectfully asserts that such excerpts do not teach sending validated program configuration data between program configuration managing computer and a managed computer, in the context claimed by applicant. Instead, Hammack only teaches a report sent to process designers. Furthermore, Hammack teaches that the "VCAT system 98 stores a text-based representation of the version control data in a file in accordance with...XML." Merely storing data as XML clearly does not meet applicant's specific claim language, namely "validated program configuration data [that] is sent to said managed computer as said XML data representation" (emphasis added).

Since at least the third element of the *prima facie* case of obviousness has not been met, a notice of allowance or a proper prior art showing of all of the claim limitations, in the context of the remaining elements, is respectfully requested.

Applicant further notes that the prior art is also deficient with respect to the dependent claims. Just by way of example, with respect to Claim 2 et al., the Examiner has relied on Col. 21, lines 33-49 in Hammack to make a prior art showing of applicant's claimed technique "wherein said program configuration data is one of: operating system registry data specifying configuration parameters of an application program; program initialization data specifying configuration parameters of an application program; and XML data directly specifying configuration parameters of an application program." Applicant respectfully asserts that Hammack only teaches "recording data indicative of modifications [with respect to] the configuration of a process" (emphasis added), and not with respect to an application program, in the context claimed by applicant.

With respect to Claim 5 et al., the Examiner has relied on Col. 21, lines 66-67 and Col. 22, lines 1-8 in Hammack to make a prior art showing of applicant's claimed technique "wherein said comparing code is part of an XML parser." Applicant respectfully asserts that such excerpts only disclose that "version control data...can be easily...parsed." Simply disclosing that data can be parsed, as in Hammack, does not meet applicant's specific claim language, namely that "comparing code is part of an XML parser" (emphasis added).

- 15 -

With respect to Claim 6 et al., the Examiner has again relied on Col. 21, lines 66-67 and Col. 22, lines 1-8 in Hammack to make a prior art showing of applicant's claimed technique "wherein said XML parser further provides validation of XML data other than said XML data representation of said program configuration data." Applicant respectfully asserts that such excerpts only relate to developing the configuration information, and do not even suggest any sort of validation, let alone validating "XML data other than said XML data representation of said program configuration data," as specifically claimed by applicant (emphasis added).

With respect to Claim 46, the Examiner has relied on Col. 6, lines 6-20 in Berg to make a prior art showing of applicant's claimed technique "wherein the map includes a mapping between keys in an operating system registry to complex data types within said DOM data representation." Applicant respectfully asserts that such excerpt does not even mention keys, let alone "a mapping between keys in an operating system registry to complex data types within said DOM data representation," as applicant claims (emphasis added).

With respect to Claim 47, the Examiner has relied on Col. 5, lines 8-15 in Berg to make a prior art showing of applicant's claimed technique "wherein values in an operating system registry are mapped to simple data types within said DOM data representation." Applicant respectfully asserts that such excerpt only generally discloses DOM nodes and an object model. Clearly, such a general disclosure does not meet applicant's specifically claimed "values in an operating system registry are mapped to simple data types within said DOM data representation" (emphasis added).

With respect to Claim 48, the Examiner has relied on Col. 12, lines 25-36 in Hammack to make a prior art showing of applicant's claimed technique "wherein an operating system registry is parsed to identify a plurality of keys and types for associating attributes with a plurality of different instances." Applicant respectfully asserts that such

- 16 -

excerpt does not even mention <u>keys</u> or that an <u>operating system registry is parsed,</u> in the specific context claimed by applicant.

Again, since at least the third element of the *prima facie* case of obviousness has not been met, a notice of allowance or a <u>proper</u> prior art showing of <u>all</u> of the claim limitations, in the context of the remaining elements, is respectfully requested.

Thus, all of the independent claims are deemed allowable. Moreover, the remaining dependent claims are further deemed allowable, in view of their dependence on such independent claims.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 505-5100. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 50-1351 (Order No. NAI1P448/02.030.01).

Respectfully submitted,
Zilka-Kotab, PC.

Kevin J Zilka
Registration No. 41,429

P.O. Box 721120
San Jose, CA 95172-1120
408-505-5100

- 16 -